

REMARKS

Claims 8, 12, 13, 20, 24 and 32-34 have been amended to address the objections raised in the Office Action. Claims 1-34 remain pending in the application.

Information Disclosure Statement:

The Office Action states that the previously submitted information disclosure statement failed to comply with 37 CFR § 1.98(a)(2), which requires a legible copy of each U.S. and foreign patent; each publication, and all other information which caused it to be listed. However, as evidenced by the return receipt postcard stamped by the Office, copies of each reference were submitted with the information disclosure statement and received by the United States Patent and Trademark Office for this application. A copy of the return receipt postcard is enclosed herewith.

Since the information disclosure statement was submitted in complete compliance with 37 CFR § 1.98 and all other applicable rules and provisions, the Examiner must fully consider the listed references and return copies of the signed and initialed statement and Form PTO-1449. A copy of the previously submitted information disclosure statement, Form PTO-1449 and date-stamped return receipt postcard are included herewith for the Examiner's convenience. If the Examiner has trouble locating the references listed on the Form PTO-1449, Applicants note that the same references were submitted in each of the co-pending applications listed on p. 2 of the information disclosure statement. The Examiner should be able to located the references in any of those co-pending application files.

Section 102(e) Rejection:

The Office Action rejected claims 1, 10, 11, 25, 28, 29 and 31 under 35 U.S.C. § 102(e) as being anticipated by Carre (U.S. Patent 6,282,579). Applicants respectfully traverse this rejection in light of the following remarks.

The Examiner states that Carre teaches a method for mapping managed object metadata comprising inputting a first data type from a first set of data types, wherein the first set of data types is expressed in an abstract syntax notation, and wherein the abstract syntax notation comprises a language for describing data; determining a corresponding second address type from a second set of data types wherein the second set of address types is expressed in an interface definition language, wherein the interface definition language comprises a language for implanting interfaces to managed objects, wherein the interface definition language is operable across a plurality of platforms and across a plurality of programming languages, and wherein the interface definition language is class independent; and returning the second address type. Applicants respectfully disagree.

Carre teaches a method for converting an address type with an address value according to a first addressing mode into a correspondent address type in another specification language. (Carre, Abstract). Specifically, Carre teaches a method for converting an OSI logic address with an address type defined in ASN.1 into a correspondent CORBA physical address with an address type defined in IDL. (Carre, col. 1 lines 35 – 56).

Accordingly, Applicants reject the Examiner's assertion that "[an] address type is a data type". As mentioned in Carre, an address type refers to how address information is calculated (e.g. a logical address or a physical address). An address type has nothing to do with the data type for the data stored at the address. For example, a given address may be represented according to a particular address type. However, the address and address type are only used to locate the data and have no bearing on the data type. The data stored at the address could be of any data type (e.g. character string, Boolean, integer, etc.). Applicants can find no language in Carre which teaches or suggests a method comprising **"inputting a first data type from a first set of data types", "determining a corresponding second data type from a second set of data types," and "returning the second data type."**

Independent claims 25 and 31 recite features similar to claim 1. As such, claims 25 and 31, along with dependent claims 28 – 30 and 32, are also patentably distinct over the cited reference for at least the reasons given above in regard to claim 1.

Claims 13, 22, 24 and 33 were rejected under 35 U.S.C. § 102(e) as being anticipated by Goldberg et al. (U.S. Patent 6,496,833) (hereinafter “Goldberg”). Applicants respectfully traverse this rejection in light of the following remarks.

The Examiner states that Goldberg teaches a method for mapping managed object metadata comprising inputting a first data type from a first set of data types, wherein the first set of data types is expressed in an interface definition language, wherein the interface definition language comprises a language for implanting interfaces to managed objects, wherein the interface definition language is operable across a plurality of platforms and across a plurality of programming languages, and wherein the interface definition language is operable to provide a mapping which is applicable to any managed object class; determining a corresponding second address type from a second set of data types wherein the second set of address types is expressed in an abstract syntax notation, and wherein the abstract syntax notation comprises a language for describing data; and returning the second address type. Applicants respectfully disagree.

Goldberg teaches a query object tool which is used to **generate** interface definitions and source code which implement a database query object. (Abstract). Specifically, Goldberg teaches query object generator tool which generates a query object server implementation and a query object IDL code 625. (col. 11, lines 21 – 28).

Applicants can find no language in Goldberg which teaches or suggests a method wherein **“the first set of data types is expressed in an interface definition language, wherein the interface definition language comprises a language for implementing interfaces to managed objects,** wherein the interface definition language is operable across a plurality of platforms and across a plurality of programming languages, and

wherein the interface definition language is operable to provide a mapping which is applicable to any managed object class,” as recited in Applicants’ claim 13.

Furthermore, Applicants can find no language in Goldberg which teaches or suggests a method comprising “**determining a corresponding second data type from a second set of data types**, wherein the second set of data types is expressed in an abstract syntax notation, and wherein the abstract syntax notation comprises a language for describing data,” and “**returning the second data type**,” as recited in Applicants’ claim 13. Applicants reject the Examiner’s assertion that QueryInfo class 702 returns a second data type express in abstract syntax notation. This in not taught by Goldberg.

Accordingly, claim 13, along with its dependent claims, are believed to patentably distinguish over the cited reference.

Independent claim 33 recites features similar to claim 13. As such, claims 33 along with dependent claim 34, are also believed to patentably distinguish over the cited reference for at least the reasons given above in regard to claim 13.

Section 103(a) Rejection:

The Office Action rejected claims 2-6, 12, 26 and 27 under 35 U.S.C. § 103(a) as being unpatentable over Carre, further in view of AAPA. Claims 14-18 were rejected under 35 U.S.C. § 103(e) as being unpatentable over Goldberg, and further in view of AAPA. These claims are patentable for at least the reason given above in regard to their respective independent claims. Furthermore, since Carre is concerned with address translation (not data type conversion), the Examiner’s combination of Carre and AAPA is improper. The combination of Goldberg and AAPA is similarly flawed.

Claims 7-9, 30 and 32 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Carre, further in view of Adusumilli (U.S. Patent 5,870,749). Claims 19-21, 23 and 34 were rejected under 35 U.S.C. § 103(a) as being unpatentable over

Goldberg, and further in view of Adusumilli. These claims are patentable for at least the reason given above in regard to their respective independent claims. Applicants additionally traverse these rejections for the following reasons. The Examiner states that Adusumilli teaches that data types in the CMIP using ASN.1 can be translated to/from custom designed data structure, wherein primitive data type and object data type are used in the abstract syntax notation and the generic primitive data type, and choice structure comprising a selector, value fields, and various data types are used in the interface definition language. The Examiner further states that it would have been obvious to have incorporated the above structure related mappings into Carre and AAPA's method/system because it would make it possible to use the very popular languages such as C and C++ for specifying a managed object in Carre and AAPA's system and resolving the incompatibility issues that exist between different specification languages. The Examiner makes a similar argument regarding the combination of Adusumilli and Goldberg.

Applicants' disagree with the Examiner's characterization of C, as used by Adusumilli, as an interface definition language on pages 8 and 10 of the Office Action. C is a *programming language*, and as such is not "**an interface definition language**, wherein the interface definition language comprises a language for implementing interfaces to managed objects, **wherein the interface definition language is operable** across a plurality of platforms and **across a plurality of programming languages**, and wherein the interface definition language is operable to provide a mapping which is applicable to any managed object class," as recited in Applicants' revised claims 1 and 13. Accordingly, claims 7 – 9, 19 – 21, 32 and 34 are also believed to patentably distinguish over the cited reference for at least the reasons given above.

CONCLUSION

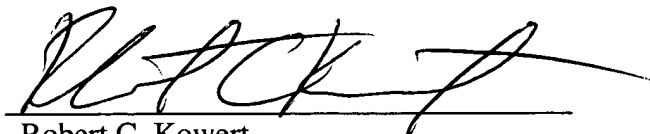
Applicant submits the application is in condition for allowance, and notice to that effect is respectfully requested.

If any extension of time (under 37 C.F.R. § 1.136) is necessary to prevent the above referenced application from becoming abandoned, Applicant hereby petitions for such extension. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5500-61600/RCK.

Also enclosed herewith are the following items:

- ☒ Return Receipt Postcard
- ☐ Petition for Extension of Time
- ☐ Request for Approval of Drawing Changes
- ☐ Notice of Change of Address
- ☐ Fee Authorization Form authorizing a deposit account debit in the amount of \$
for fees ().
- ☒ Copy of previously submitted information disclosure statement, Form PTO-1449 and
date-stamped return receipt postcard.

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: August 7, 2003